ParaNames: A Massively Multilingual Entity Name Corpus

Jonne Sälevä and **Constantine Lignos**
Michtom School of Computer Science
Brandeis University
{jonnesaleva,lignos}@brandeis.edu

Abstract

We present ParaNames, a Wikidata-derived multilingual parallel name resource consisting of over 118 million names for 13.7 million entities, spanning over 400 languages. ParaNames is useful for multilingual language processing, both for defining name translation tasks and as supplementary data for other tasks. We demonstrate an application of ParaNames by training a multilingual model for canonical name translation to and from English.

1 Introduction and Related Work

Our goal for ParaNames is to introduce a massively multilingual entity name resource that provides names for diverse entities in the largest possible set of languages and can be kept up to date through a mostly-automated preprocessing procedure. In this extended abstract, we summarize our approach to transforming the Wikidata knowledge graph into a set of parallel entity names identified with the high-level types of person, location, and organization.

We do not claim to be the first to harvest the parallel entity names available from Wikidata or Wikipedia. There is scattered prior work in this area, with one of the earliest explorations at scale being performed by Irvine et al. (2010). Recently, Benites et al. (2020) used Wikipedia as a data source and automatically extracted potential transliteration pairs, combining their outputs with several previously published corpora into an aggregate corpus of 1.6 million names.

2 Constructing the resource

To construct our dataset, we began by extracting all entity records from Wikidata and ingesting them into a MongoDB instance. Each entity in Wikidata is associated with several types of metadata, including names for it across languages. Given that we are working with such a large-scale dataset, there are important challenges that arise.

Script usage While language codes can identify a specific script for a language, many Wikidata labels do not conform to the scripts used by each language. In many cases, this is simply a data quality issue, such as with Greek where approximately 8.9% of ORG entities are written in Latin script.

However, in other cases, the presence of several scripts can also reflect real variation in the citation forms used in the language, as many languages (e.g. Kazakh) commonly use several scripts. While we explored automated methods of identifying names in incorrect scripts, we decided that manually constructing a list of allowed scripts for each language would yield the best results. We used Wikipedia as an authoritative source to look up which scripts are used to write each language, and filtered out all names whose most common Unicode script property is not among the allowed ones.

Providing entity types Downstream tasks and analysis of performance across different entity types often require that entities have a single high-level type. Wikidata has a complex type hierarchy, but we infer simpler entity types for as many entities as possible. We identified suitable high-level Wikidata types—Q5 (human) for PER, Q82794 (geographic region) for LOC, and Q43229 (organization) for ORG—and classified each Wikidata entity that is an instance of these types as the corresponding named entity type. In total, our resource includes 8,726,033 PER entities, 3,078,428 LOC entities and 2,196,035 ORG entities.

3 Experiments

To demonstrate an application of ParaNames, we train multilingual Transformer-based models that map entity name from English to one of Arabic, Armenian, Georgian, Greek, Hebrew, Japanese, Kazakh, Korean, Latvian, Lithuanian, Persian (Farsi), Russian, Swedish, Tajik, Thai, Vietnamese, and Urdu and vice versa. We chose these languages

Proceedings of the 4th Workshop on Computational Typology and Multilingual NLP (SIGTYP 2022), pages 103 - 105
July 14, 2022 ©2022 Association for Computational Linguistics

Language	Accuracy	CER	F1
Swedish	88.25 ± .02	0.08 ± .00	97.15 ± .01
Vietnamese	80.75 ± .02	0.17 ± .00	94.08 ± .01
Latvian	67.86 ± .02	0.14 ± .00	95.19 ± .01
Kazakh	55.38 ± .04	0.16 ± .00	93.93 ± .01
Tajik	49.62 ± .05	0.20 ± .00	92.77 ± .01
Lithuanian	47.39 ± .03	0.28 ± .00	89.53 ± .01
Thai	43.94 ± .05	0.29 ± .00	89.91 ± .01
Armenian	39.92 ± .05	0.28 ± .00	90.04 ± .01
Georgian	34.44 ± .02	0.29 ± .00	89.29 ± .01
Korean	33.27 ± .05	0.32 ± .00	88.46 ± .01
Russian	32.81 ± .06	0.38 ± .00	84.80 ± .02
Urdu	31.92 ± .03	0.23 ± .00	91.48 ± .01
Japanese	29.00 ± .04	0.33 ± .00	87.79 ± .01
Persian	28.68 ± .05	0.28 ± .00	89.84 ± .02
Arabic	25.74 ± .03	0.32 ± .00	89.23 ± .01
Greek	24.70 ± .03	0.35 ± .00	86.60 ± .01
Hebrew	15.24 ± .07	0.44 ± .00	84.58 ± .02
Overall	42.88 ± .02	0.27 ± .00	90.27 ± .01

Table 1: Canonical name translation performance for the X → En task, computed on the test set using our baseline configuration with language special tokens on the source side.

Language	Accuracy	CER	F1
Swedish	85.60 ± .04	0.10 ± .00	96.11 ± .02
Vietnamese	48.86 ± .01	0.35 ± .00	82.87 ± .01
Latvian	69.28 ± .07	0.13 ± .00	95.49 ± .01
Kazakh	58.69 ± .09	0.14 ± .00	94.85 ± .02
Tajik	54.38 ± .02	0.18 ± .00	93.82 ± .02
Lithuanian	50.76 ± .09	0.23 ± .00	91.61 ± .03
Thai	14.80 ± .04	0.42 ± .00	83.01 ± .02
Armenian	50.45 ± .05	0.22 ± .00	92.41 ± .01
Georgian	51.82 ± .04	0.22 ± .00	92.56 ± .01
Korean	38.63 ± .05	0.33 ± .00	88.18 ± .01
Russian	44.59 ± .04	0.33 ± .00	89.81 ± .02
Urdu	14.14 ± .08	0.45 ± .00	80.74 ± .03
Japanese	28.70 ± .01	0.42 ± .00	84.42 ± .02
Persian	22.90 ± .05	0.41 ± .00	81.64 ± .05
Arabic	41.70 ± .02	0.28 ± .00	89.40 ± .01
Greek	29.67 ± .06	0.36 ± .00	86.88 ± .01
Hebrew	35.71 ± .03	0.34 ± .00	88.16 ± .01
Overall	43.57 ± .02	0.29 ± .00	88.94 ± .01

Table 2: Canonical name translation performance for the En → X task, computed on the test set using our baseline configuration with language special tokens on the source side.

as they cover a wide geographic distribution, as well as several different orthographic systems, language families and typological features.

To create the parallel data, we extracted all entities that had names in English and at least one of the selected languages and split them into train, dev, and test sets using an 80/10/10 split. We also added "special tokens" to the beginning of each input to provide the model with additional information, e.g. entity type (`<PER>`), language of non-English label (`<kk>`) and/or its script (`<Cyrillic>`).

We use a single NVIDIA RTX 3090 GPU for training and decoding, and train our model for up to 90k updates using Adam.[1] We evaluate using three metrics: accuracy, mean F1-score (Chen et al., 2018), and character error rate (CER).

As our first experiment, we trained our models with only a language special token on the source side. The results in both translation directions can be seen in Tables 1 and 2. When translating to English, our model performs best on Swedish, Vietnamese and Latvian, which is unsurprising as all use the Latin script. However, Latvian names tend to be more inflected and generally match English less often, which explains its lower ranking. Kazakh and Tajik follow next, which also makes sense as Cyrillic can be transliterated to Latin script relatively unambiguously. Model performance is consistently worst on Hebrew—most likely caused by the lack of vowels in the Hebrew names, which the model must infer when translating to English.

When translating from English, the model performs best on languages similar to when translating to English. Swedish and Latvian have the highest accuracy, followed by Kazakh, Tajik, and Georgian. For Hebrew, the model performs much better; a potential explanation for this is the lack of vowel diacritics. Interestingly, the reverse is true for Thai, where the model performs less than half as accurately as when translating into English.

We also hypothesized that incorporating other information could be helpful, and repeated the experiment using a mixture of language, type token, and script special tokens. Overall, the results within each language tended to be quite similar regardless of tokens. The best settings were to use all three special tokens when translating from English, and language and type tokens when translating to English. While small, the differences from baseline were statistically significant for almost all settings.

4 Conclusion

ParaNames supports the modeling of parallel names for millions of entities in over 400 languages. It can enable multifaceted research in names, including name translation/transliteration and further research in named entity recognition and linking, especially in lower-resourced languages.

[1] Other hyperparameter values are nearly identical to the best configuration in Moran and Lignos (2020).

References

Fernando Benites, Gilbert François Duivesteijn, Pius von Däniken, and Mark Cieliebak. 2020. TRANSLIT: A large-scale name transliteration resource. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 3265–3271, Marseille, France. European Language Resources Association.

Nancy Chen, Xiangyu Duan, Min Zhang, Rafael E. Banchs, and Haizhou Li. 2018. NEWS 2018 whitepaper. In *Proceedings of the Seventh Named Entities Workshop*, pages 47–54, Melbourne, Australia. Association for Computational Linguistics.

Ann Irvine, Chris Callison-Burch, and Alexandre Klementiev. 2010. Transliterating from all languages. In *Proceedings of the 9th Conference of the Association for Machine Translation in the Americas: Research Papers*, Denver, Colorado, USA. Association for Machine Translation in the Americas.

Molly Moran and Constantine Lignos. 2020. Effective architectures for low resource multilingual named entity transliteration. In *Proceedings of the 3rd Workshop on Technologies for MT of Low Resource Languages*, pages 79–86, Suzhou, China. Association for Computational Linguistics.